

APL+Win v10.3 Update
Copyright (c) 2010 APLNow LLC.
All Rights Reserved
Dec. 29, 2010

This update is recommended for all current APL+Win subscribers that have version 10.0, 10.1 or 10.2 installed. Current APL+Win subscribers can download the update for free at <http://www.apl2000.com/software.php>.

Update description:

This update release includes the following important new enhancements in version 10.3:

Enhancement #1: Dynamic control of Event Stops in the session

A new INI file setting named [Session]DefaultES=n allows specification of the default Event Stop setting for a session.

A new [Save] button has also been added to the Event Stops dialog that saves the current setting as the new defaults. It prompts for confirmation that you want to save the current settings as the default, and if confirmed it directly saves the [Session]DefaultES state immediately without waiting for end of session saving or Save Settings Now (which do NOT save Event Stop defaults). Also note that the default settings do NOT apply to sessions started in runtime system (RTS) or to the WSEngine COM server settings.

The [Session]DefaultES@Server setting controls the default state of event stops for WSEngine server session. This is the setting that gets saved by the dialog when it is used in a COM server session (or it can be set manually).

Note 1: [Session]DefaultES applies only to normal non-runtime sessions

Note 2: [Session]DefaultES@Server applies only to COM server sessions

Note 3: Event stops for runtime sessions are always zero (0).

Also added □SYS[32] to report the current Event Stop settings. This can be assigned to change the settings dynamically. The only setting allowed in an runtime session is 0. Any settings outside the acceptable range are ignored such that only valid bits get set.

The following is a summary of the Event Stop bits that can be coded in decimal (and hex). These codes correspond to the check boxes that can be selected in the Event Stops dialog. Subordinate

items (indented below a higher level item) are disabled if their parent is not selected. For example, but 1 must be selected in order for any other settings to be enabled. Similarly, bit 16 must be selected in order for bits 32, 64, or 128 to take effect.

1	(0x00000001)	&Enable Event Stops
8	(0x00000008)	&SUPPRESS error handling at suspension (enable in called fns)
16	(0x00000010)	&IMMEDIATE execution from suspended context
32	(0x00000020)	Suspend before &handling errors
64	(0x00000040)	Suspend before □ERROR e&xits function
128	(0x00000080)	Suspend before &branching from □ELX handler
256	(0x00000100)	&LOCAL stepping context (where stepping started)
512	(0x00000200)	Suspend before &handling errors
1024	(0x00000400)	Suspend before □ERROR e&xits function
2048	(0x00000800)	Suspend before &branching from □ELX handler
4096	(0x00001000)	&REMOTE stepping context (functions called by local context)
8192	(0x00002000)	Suspend before &handling errors
16384	(0x00004000)	Suspend before □ERROR e&xits function
32768	(0x00008000)	Suspend before &branching from □ELX handler
65536	(0x00010000)	&CALLBACK Context
131072	(0x00020000)	Suspend before &handling errors
262144	(0x00040000)	Suspend before □ERROR e&xits function
524288	(0x00080000)	Suspend before &branching
978937	(0x000EEFF9)	THE DEFAULT VALUE if not specified by INI file override

Enhancement #2: Support multi-level Undo/Redo in the Session Manager function and character editor sessions

This enhancement also adds the following:

1. the Redo menu item and corresponding shortcut (Ctrl+Y) in the Edit menu.
2. the Undo and Redo toolbar shortcuts in the Standard Toolbar.
3. the capability to resume pending edit session(s) after an unanticipated failure in the APL+Win development system. When APL+Win is restarted, a dialog prompt appears with the following three choices:
 - a. Click YES to resume previous edit sessions.
 - b. Click NO to discard the previous edit sessions.
 - c. Click CANCEL to leave previous edit sessions pending.

Enhancement #3: Support UCS characters in Draw method's Text operation and Printer object's Print method

The Draw method's Text operation has been modified to understand UCS character in exactly the same way as it previously understood integer arguments. However, unlike integers, UCS characters can be used alone without specifying coordinates like this:

```
OWI 'Draw' ('Text' (UCS 97 98 99))
```

in the same way as it previously understood character data:

```
OWI 'Draw' ('Text' 'abc')
```

Whereas with integer data you must specify coordinates such as this:

```
OWI 'Draw' ('Text' (97 98 99) 0 0)
```

This change to the Draw method works for all object types that accept Draw method, including the Printer.

The Printer object's Print method has been enhanced to understand UCS or integer data (as UCS indexes) in the range 0-65535. This is NOT full Unicode (it's UCS).

So you can therefore print like this:

```
'printer' OWI 'Print' 'abc'
or
'printer' OWI 'Print' (UCS 97 98 99)
or
'printer' OWI 'Print' 97 98 99
```

For best results when printing UCS characters, it's recommended that you set the font property for the Printer object be a Unicode font.

=====

Please send any questions or comments to support@apl2000.com.

APL2000 Technical Support